

DAQ Overview

Mario Martínez
(FNAL)

(08/21/2002)

List of items covered in this talk:

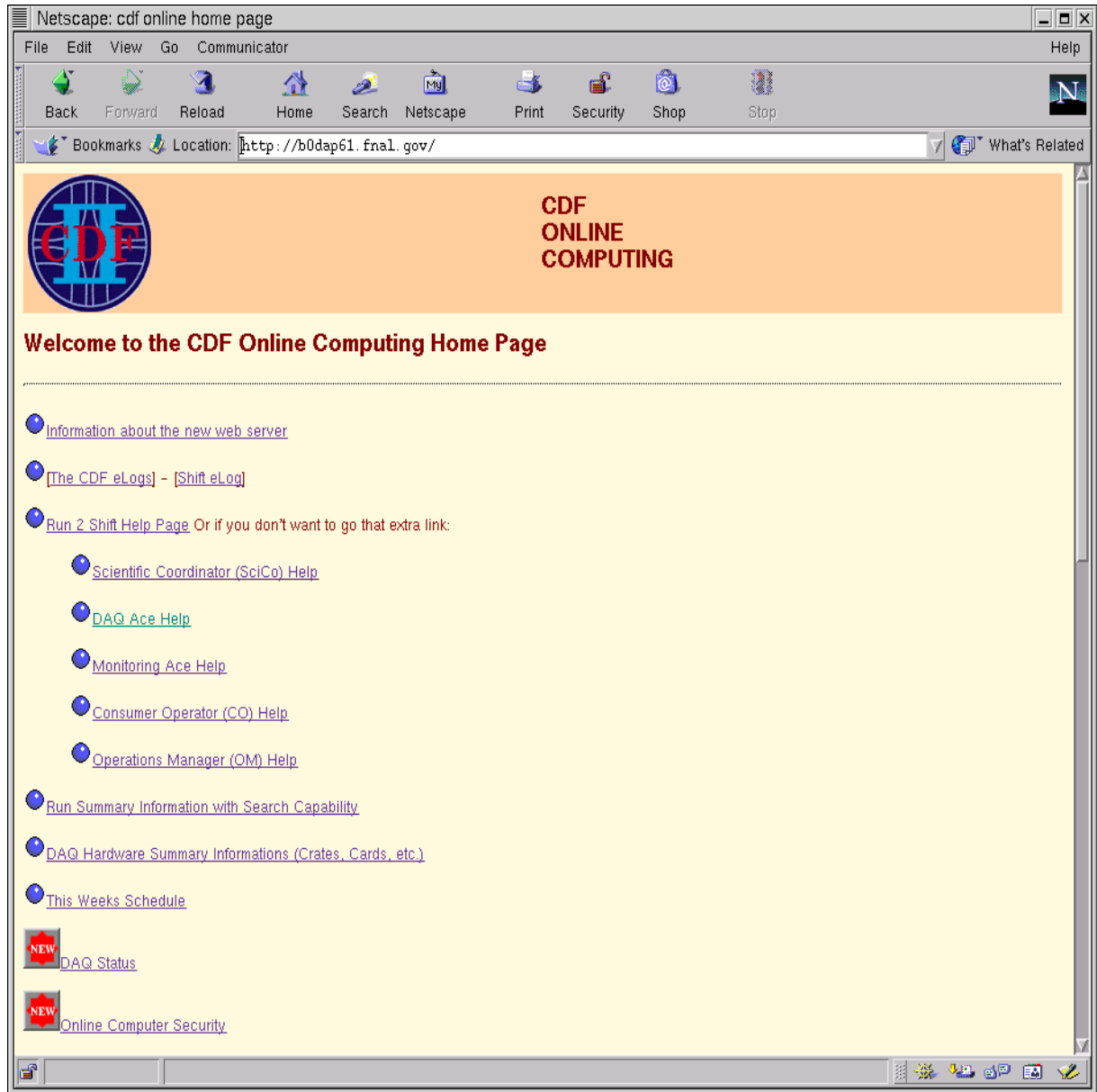
- Overview of DAQ components
- Front End crates
- Trigger System **
- Run Control **
- Dead Time + Warnings/Errors **
- DAQ Monitoring **
- Final Remarks

I will not cover Silicon DAQ (separate talks)

(I will heavily use information from previous talks)
www-cdfonline.fnal.gov/ace2help/training/

** → separate talk

ONLINE WWW PAGE

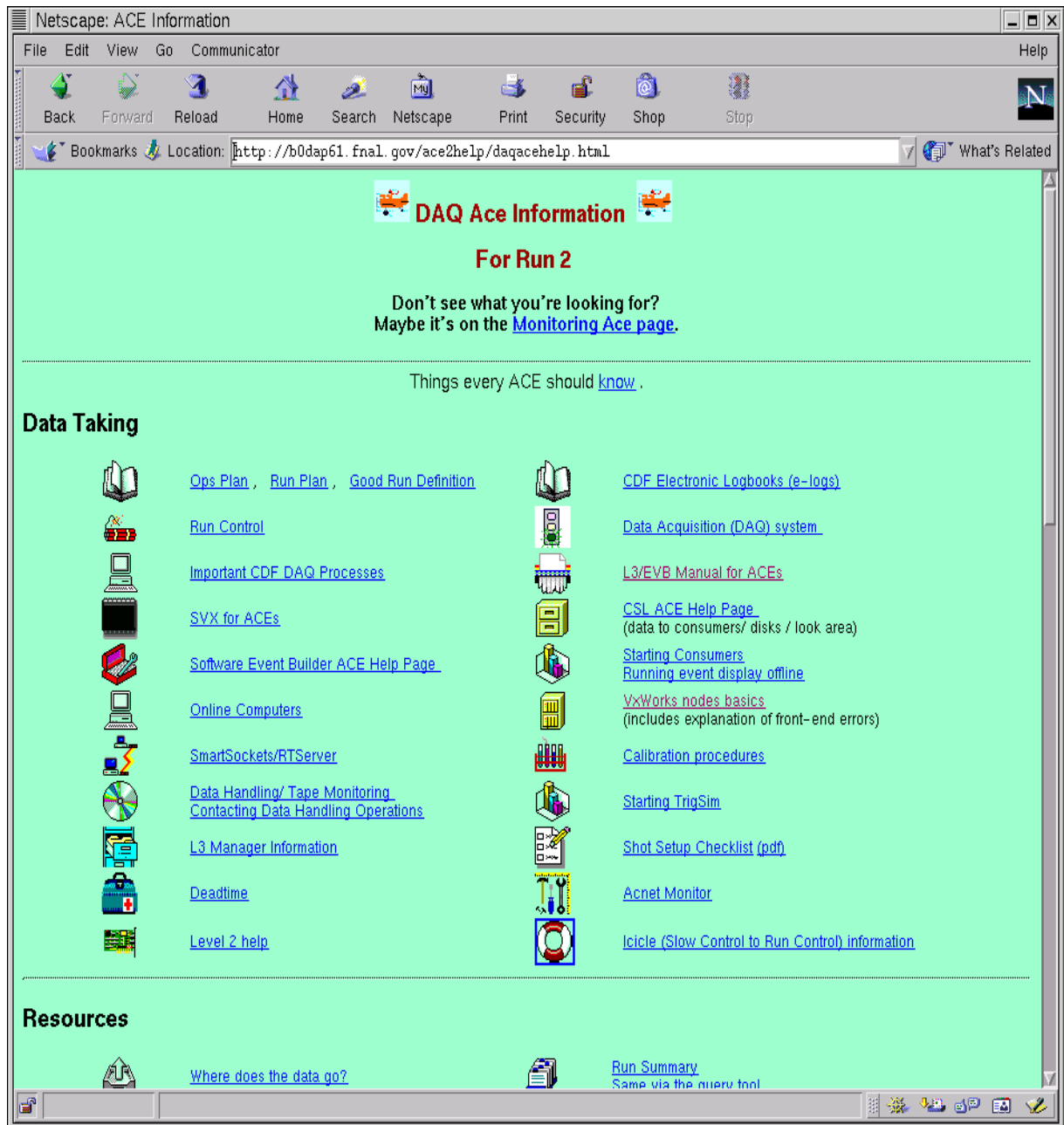


⇒ All information is there

www-cdfonline.fnal.gov/

PS. In this talk whenever you find www-b0.fnal.gov:8000 means www-cdfonline.fnal.gov

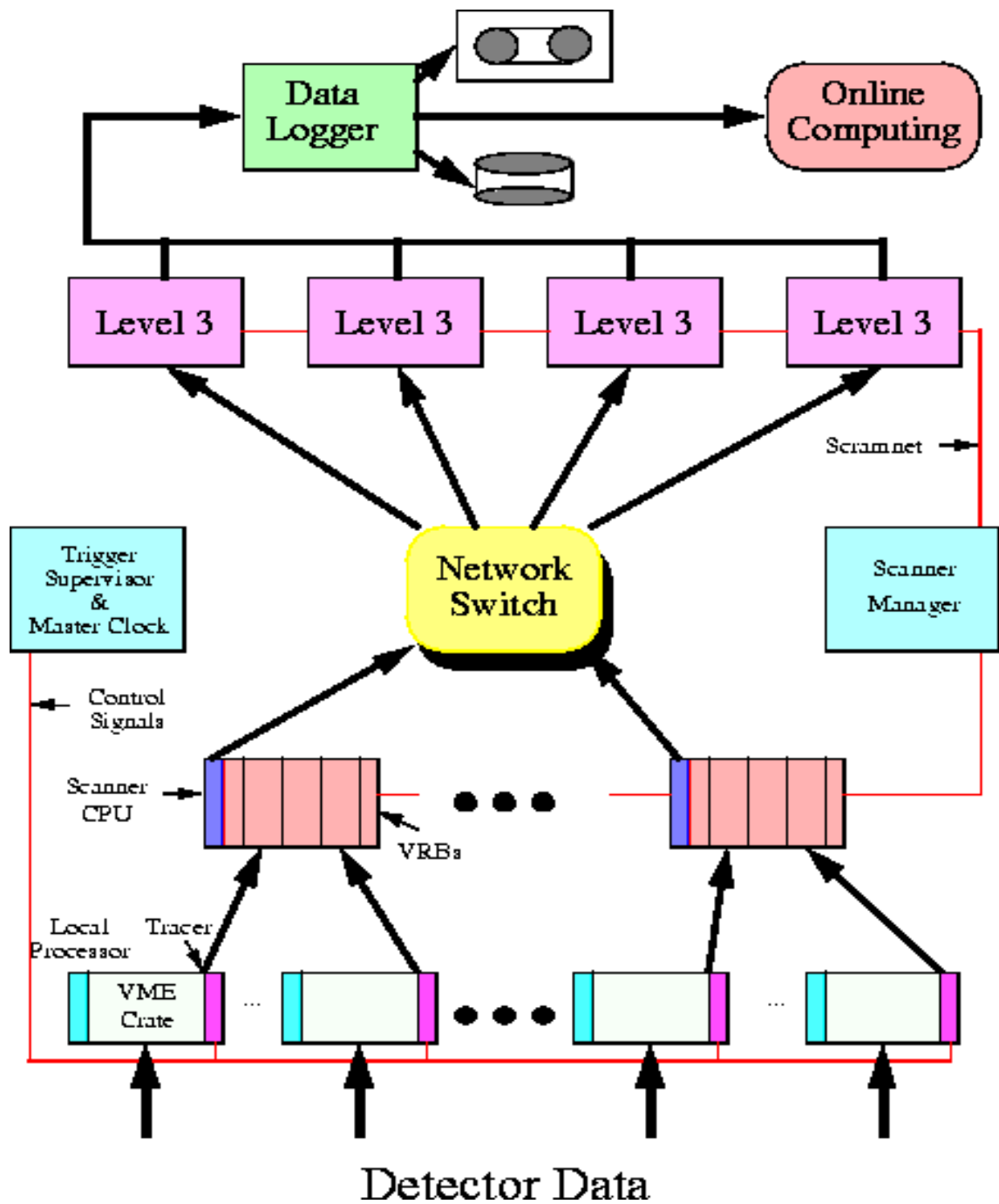
ACE WWW HELP PAGE



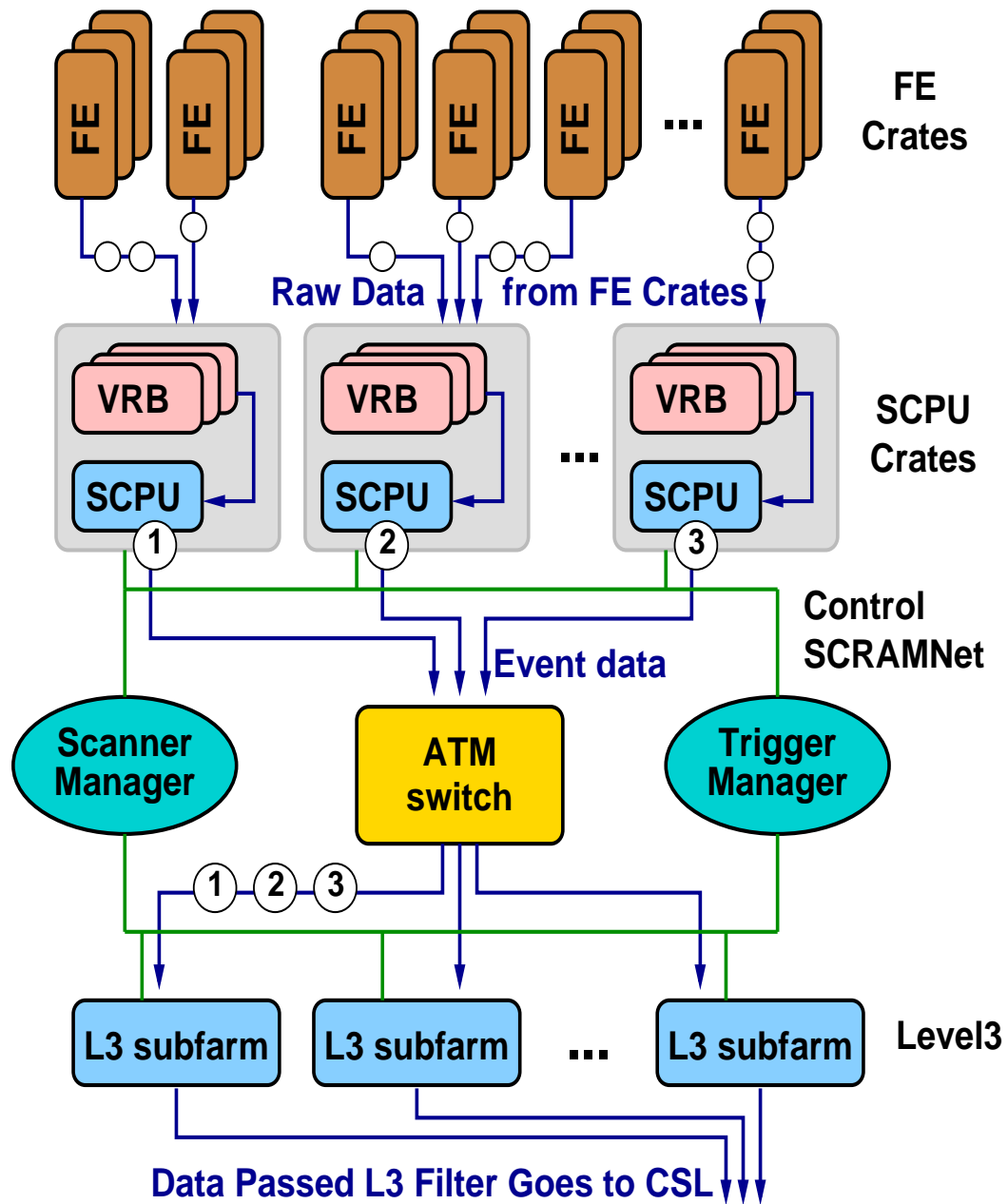
⇒ Instead of CNN ..Pais, Repubblica, Spiegel
....spend your free time here...

www-cdfonline.fnal.gov/ace2help/daqacehelp.html

Overview of DAQ Components (I)



Overview of DAQ Components (II)



DAQ Components (Brief)

- Front End (FE) and Trigger VME Crates:

- reads out detector
- formats and transfer data.

- Trigger Supervisor and Cross-points:

- receives trigger decision from L1 and L2 Global
- distributes trigger signals (via X-points) to FEs

- Event Builder:

- after L2 accept signal, receives data from FE Crates
- assembles event fragments into one block
- sends the data to the L3 farm.

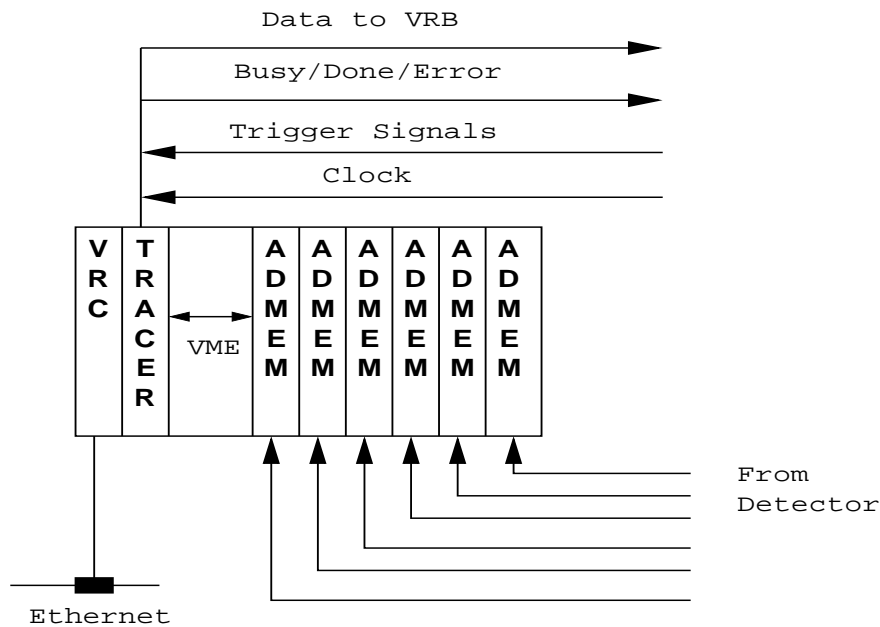
- Level 3 Trigger:

- formats the data into the final format (root)
- has access to complete event
- runs offline code to make a trigger decision.

- Consumer Server/Logger:

- receives data from L3
- writes data to disk (different streams)
- sends event to consumes

Typical Front End/Trigger Crate



- FE cards (TDCs, ADMEMs...)
- VME Readout Controller (VRC):
 - Motorola MVME 2301 with Power PC 603 CPU running VxWorks (real-time operating system)
 - forms a mini-bank attaching header to the FE data
 - sends back “DONE” signal to trigger supervisor.
- TRACER:
 - used to fan out trigger and clock signals to VME
 - transports data out of the crate and FE cards

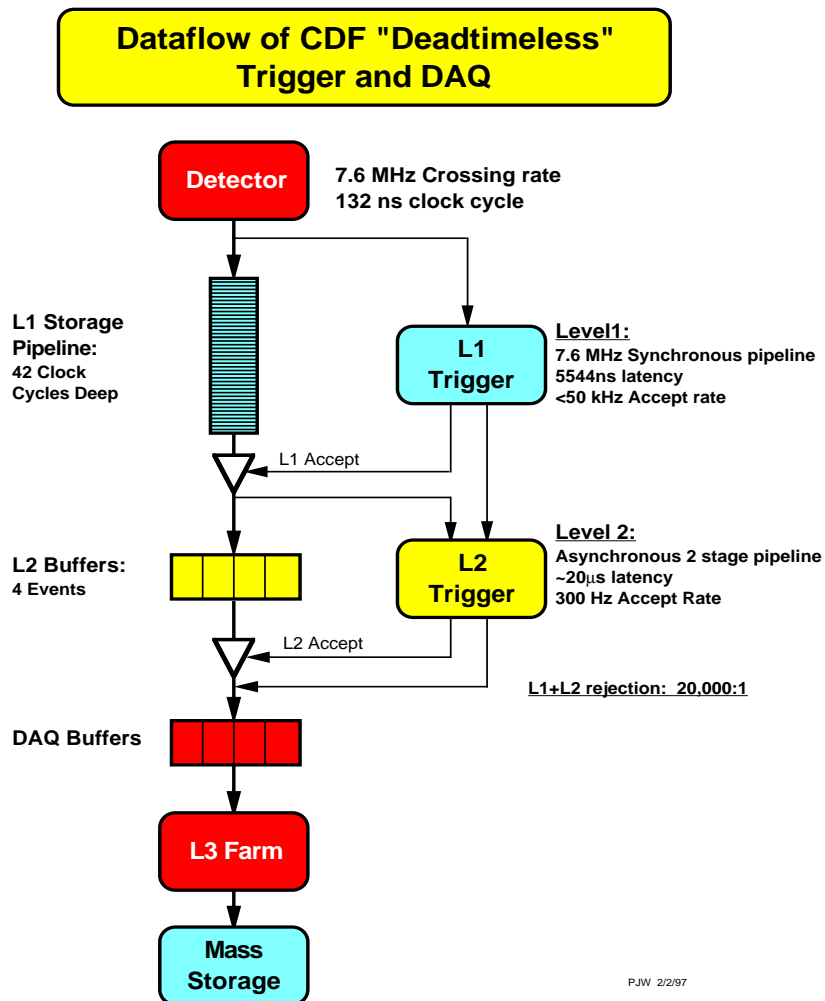
ACE access to Crates

- FE crates have node names such as b0ccal00 ..ect...
- You can log in to the crate to check status (useful for tracking problems):
 - setup fer
 - vxlogin b0ccal00
- You can also access the crate using minicom (serial line) from b0dap10 machine (if network connection to VRC is not available)
 - telnet b0dap10
 - setup fer
 - minicom b0ccal00
- You can use the reset lines in the crate ... BUT only if all the rest did not work.

⇒ Follow instructions of [www page:
b0dap61.fnal.gov/vxworks/vxworks.html](http://www.b0dap61.fnal.gov/vxworks/vxworks.html)

Trigger System

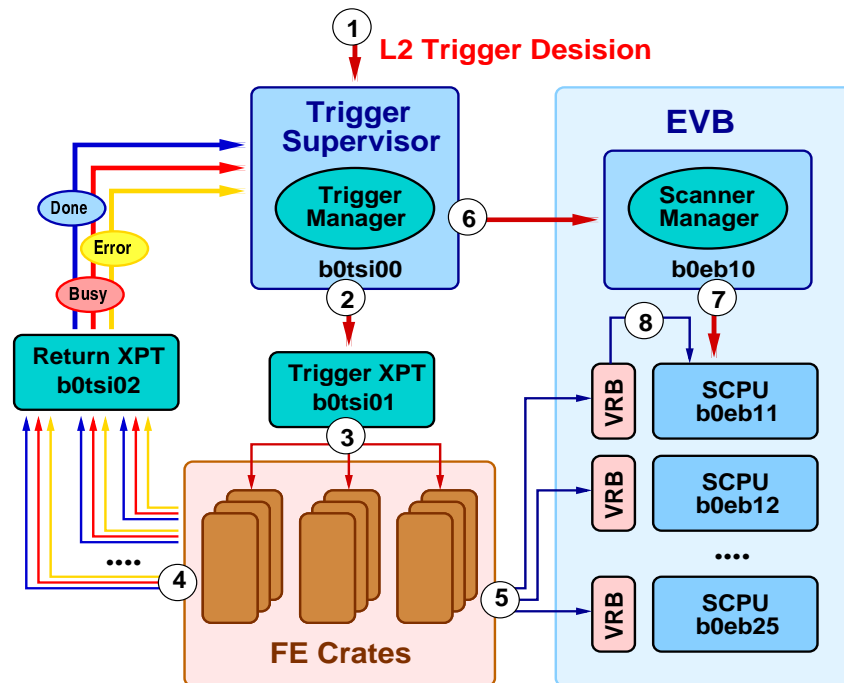
The trigger system is used to select an event rate of 75 Hz from the 7.6 MHz (132 ns crossing) beam crossing rate.



Done in three stages: Beam crossing rate of 7.6 MHz (for 132 ns crossing) is reduced to ≤ 50 KHz, by the L1 trigger, reduced to 300 Hz by L2 and finally to about 75 Hz by L3.

The L2 Trigger sends the trigger decision to the Trigger Supervisor (TS).

The L2 trigger decision is sent to the FE crates through the trigger cross point.



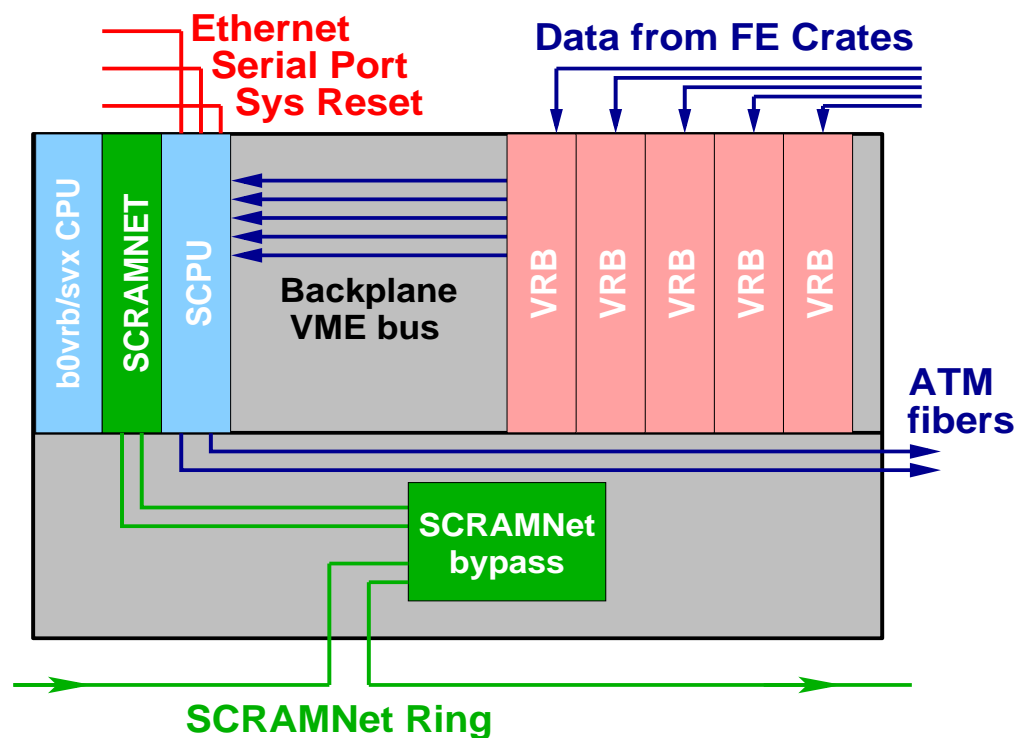
When the trigger decision is received in the FE crate, the VME Readout Controller (VRC) sends back a DONE signal to the TS via the return cross point indicating that it is ready to receive the next trigger decision.

Data is read out from the FE cards (TDCs, ADMEMs...) formatted and sent via the TRACER to the VME Readout Buffers VRBs.

If there is not enough space to write out the event to the VRB a **BUSY signal** is sent back to the TS so that the TS does not issue another trigger which leads to busy deadtime.

→ If the busy is not deasserted in time we can get a Busy Timeout causing the run to halt.

Each VRB can receive data from up to 10 different front end crates.

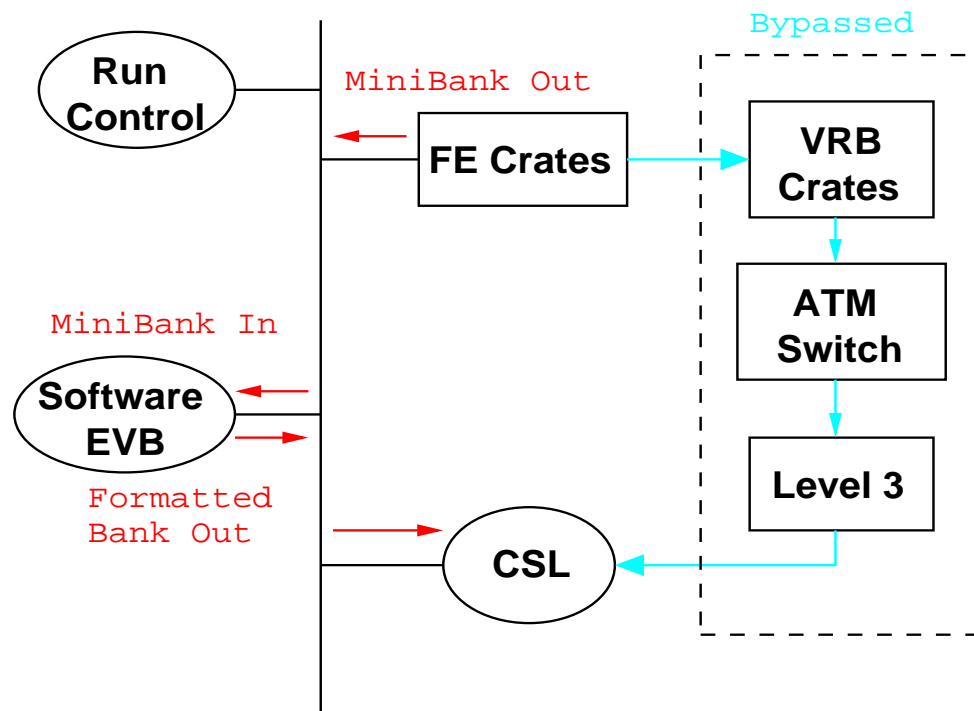


Data is readout of the VRBs by the Scanner CPU (SCPU) and sent through the ATM switch to the L3 farm.

The SCPU communicates with the TS via SCRAMNET.

Software Event Builder

FE crates can send the mini banks over ethernet to a software client that collects the event fragments and reformats them into the final data format.



Events are sent to the CSL and can be distributed to the consumers or written to disk.

Used for debugging parts of the system and for calibrations.

Depending on how much data is being read out the rates can range from a few Hz to a few tens of Hz.

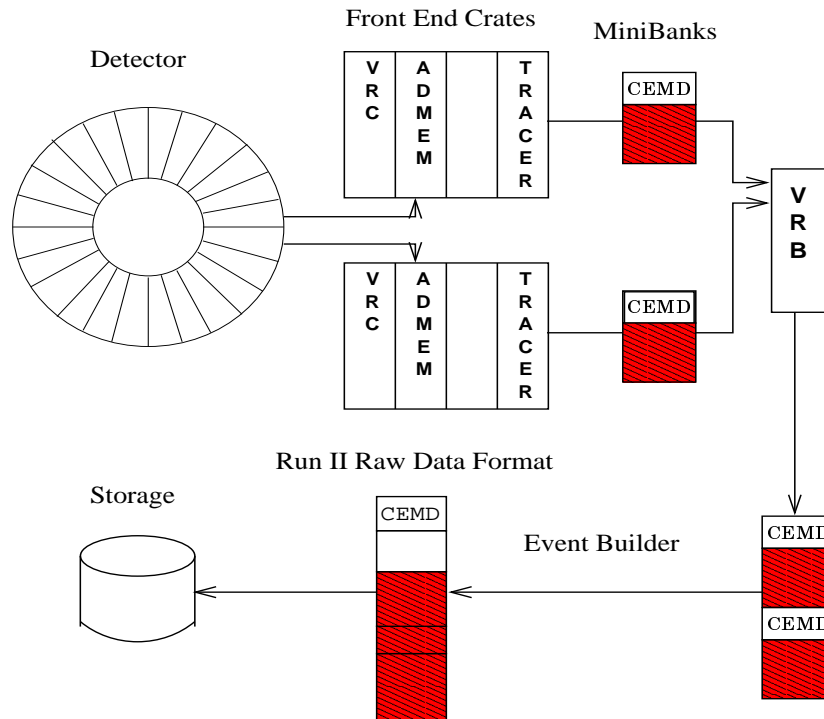
More details at:

<http://www-b0.fnal.gov:8000/ace2help/sevb/>

Data Format and Reformatter Errors

The reformatter code assembles the minibanks into the final event format making it available to the L3 analysis code.

→ Events having corrupted fragments are rejected by the reformatter.



At L3 the reconstruction can add reconstructed objects to the event record.

Events passing the L3 trigger are sent to the Consumer Server Logger and a fraction of the events are distributed to monitoring consumers.

Events are transferred to Feynmann Computing Center (FCC) for storage on tape.

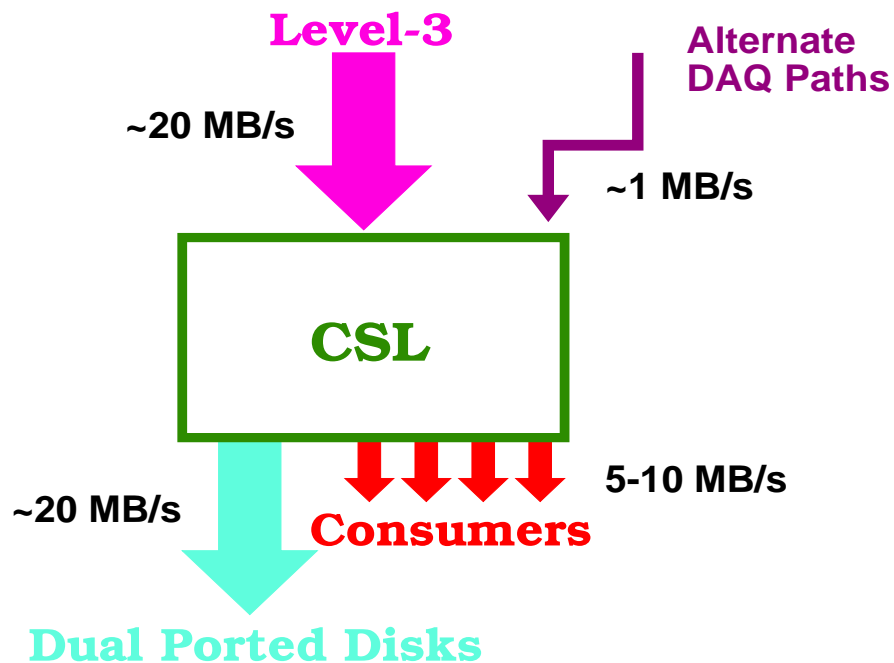
Users access the data through the Data Handling System.

CSL

If an event passes a L3 trigger it is first sent to an output node then to the Consumer Server Logger (CSL).

The CSL distributes events to the various consumers which are used to check the quality of the data and the proper functioning of the trigger system.

The CSL writes data to disk in B0 separating it into different data streams based on the L3 trigger decision.

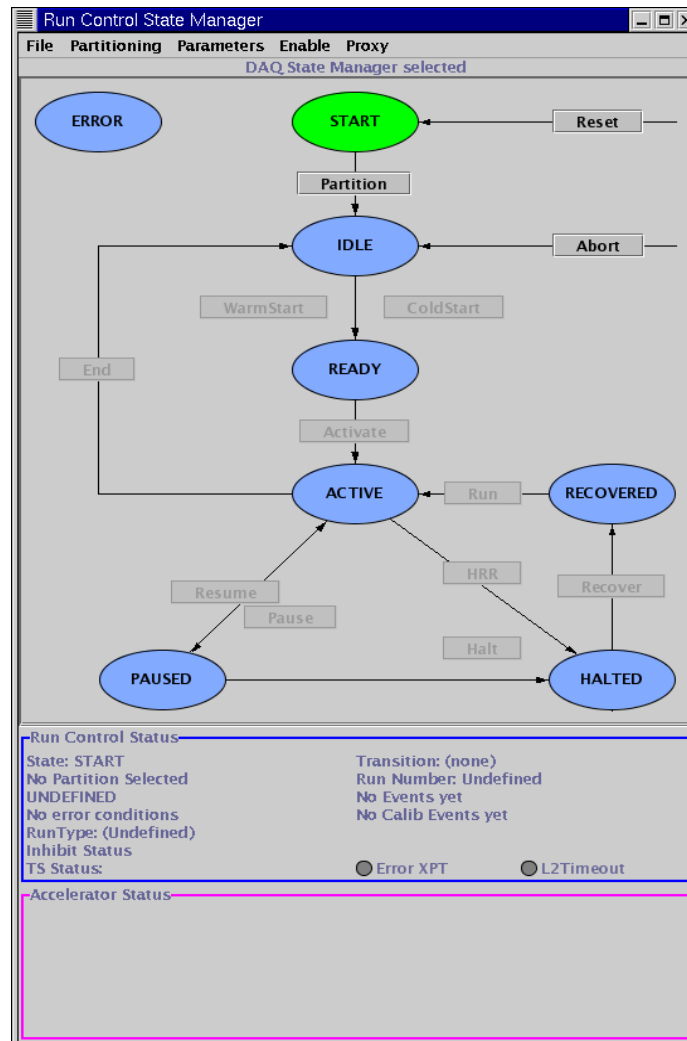


Data is copied from the disk buffers at B0 to the FCC disks then written to tape.

Run Control

Coordinates the configuration, starting and stopping of runs.

Written in Java, uses a Graphical User Interface showing a state transition diagram to control the many distributed clients.



Can group together clients into a *partition*. Can run with up to eight hardware partitions simultaneous.

Allows inclusion or exclusion of individual cards or crates, masking of bad channels...

Uses the commercial message passing software package smart-sockets.

Uses DaqMsg (layered on top of smartsockets) to provide automatic code generation to conveniently pack and unpack data structure (messages...).

Machine independent communication... clients written in Java running under Linux communicating with clients written in C running under VxWorks.

Clients subscribe to a subject and *configuration and control* messages are broadcast to all clients who are subscribed to a particular message.

Subjects have the syntax

/partition-0/frontEnd/ccal/00

Can use wildcarding to broadcast message to all clients of a certain type. Used for “run sequencing”, to bring a group of clients through a transition before a second group of clients.

All communication goes through the *rtserver*.

If the RC GUI crashes the run could still be going... and you can try reattaching a new RC to the current run.

More details in a separate talk...

Error Handler

Error messages from the different clients are sent to the Error Handler, which displays the message on the screen and also logs the error messages.

After setting up the fer package, (setup fer) the environment variables for the error handler will be set.

The location of the error log file is:

`$ERRMON_LOGDIR/errorfile136574.log`

The interpretation of the error is done by the error handler, it is centralized so that the operator has one place to look.

Alerts the user of serious error conditions. Currently an orange window appears when there is a fatal error condition. Text message instructs the operator what to do.

In addition to the visual alert there is a voice alert which states the problem.

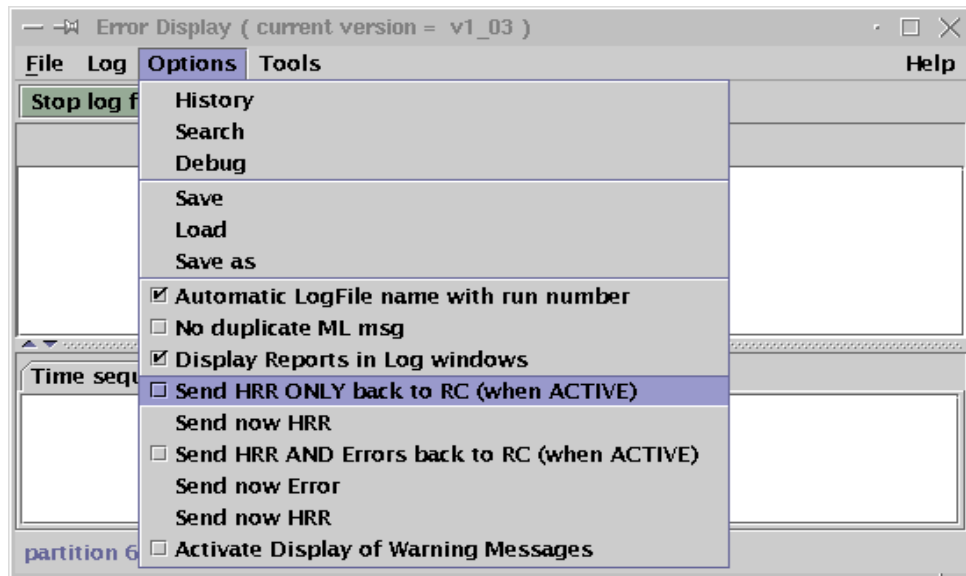
Can also be used to issue an automated Halt-Recover-Run sequence in the case of a Done or Busy timeout.

You should always be running with the automatic HRR enabled.

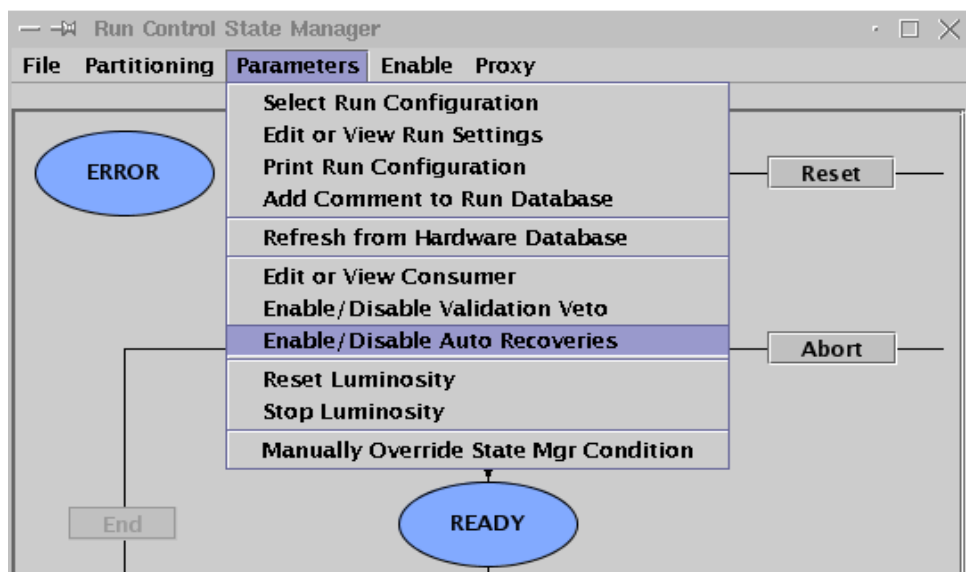
Report any problems with this feature in the shift log and send email to the RC email address:

`cdf-rc-support@fnal.gov`

You have to enable this option of sending commands from the error handler to RC.



and you have to enable RC to accept remote commands from the error handler.



Will build more intelligence into the error handler → expert system.

Dead Time

More details about the sources of deadtime and how to identify the source can be found at:

<http://www-b0.fnal.gov:8000/ace2help/deadtime.html>

The rate limit into the EVB is about 375 Hz. The logging rate limit of the CSL is 20 MB/s which corresponds to about 75 Hz.

Actual rates depend on the data volume, number of clients in the run and detector occupancy.

Normally the trigger tables are defined such that the dead time is less than 5%.

If the dead time is higher than $\sim 5\%$ then the source needs to be identified.

From the *DaqMon Rates and Deadtime* display you can see the total dead time of the system and the fraction from various sources.

The most common type of deadtime you will encounter are from “Busy” and “Readout”.

Busy

This indicates that the VRB buffers are full and cannot accept more data.

Either the L3 accept rate is too high (faulty trigger) and we are limited by the 20 MB/s CSL rate, it is taking too long to process events at L3 or it is taking too long to read in the events or L3.

If the L3 display (one of the DAQ monitors) is “mostly green” this indicates that the processors are mostly occupied by trying to *output* the events to the CSL.

Check the CSL logging rate, if it is around 20 MB/s it may indicate that a L1 or L2 trigger is firing at too high of a rate.

One of the disks that the CSL buffers data onto is a slow disk and when writing to it we see logging rates of 15 MB/s.

A L3 display that is mostly “dark blue” indicates that the processors are busy *processing* the event. So far we have not been limited by the processing capacity of L3.

A L3 display that is mostly “light blue” indicates that the processors are busy *inputting* events. A BUSY for this case may occur if the event size is very large, for example noisy channels can lead to large events...

Readout

Readout deadtime occurs when the FE processors are taking too long to readout the event.

Many systems have a fixed data size, however for some the data volume increases with increasing luminosity.

For the TDCs the DSP processing time also increases with the number of “hits” for a channel.

→ A typical source of readout deadtime is high occupancies for the TDCs which occurs when some channels are oscillating resulting in many “hits”. The TDC DSP cannot process events fast enough...

One can identify the “bad guy” by using the RXPT monitor to see which crate shows up as the last to return DONE.

L2 Deadtime

L2 or Readout Deadtime

At a L1 accept rate of about 3.5 KHz and a L2 accept rate of about 250 Hz we have seen a dead time of about 2% due to a combination of “L2” and “L2 or Readout”.

This is believed to be the result of the way buffers are managed by L2 which effectively reduces the number of available buffers from four to three.

Typical Warnings/Errors

Warning: COT Truncated Data

For very high occupancy events or when there is noise on a channel the data coming from one of the COT crates can be larger than what can be held in the VRB buffer.

In this case we truncate the data and you will get an error message of the type:

```
(MLE) b0cot14:5:37:25 AM->Runtime Error 1, Event 4793: data truncated  
(MLE) b0cot02:5:54:23 AM->Runtime Error 2, Event 53148: data truncated
```

Warning: Bunch counter mismatch

Each front end card is checked that the BC is consistent. If there is a mismatch this warning will be sent.

For some crates this is a known problem and we do not try to go through a HRR sequence since the problem is cleared on the next event.

In other crates this problem is serious and will result in a done timeout. In this case we do issue a HRR to recover.

Warning: Reformatter Errors

Events with corrupted data fragments cannot be assembled into an event and are rejected by the reformatter process.

If the instantaneous rate of reformatter errors (measured over 30 seconds) is greater than 1% the error handler will pop up a warning message.

In this case follow the instructions on the window.

Reformatter errors are usually the result of corrupted silicon data and if they persist the offending wedge may have to be removed by an expert.

Error: Done Timeout

If an error is detected in the front end crate the process may not set the “done” bit generating a done timeout.

The run can be recovered by going through the Halt Recover Run sequence.

→ This type of error is detected by the error handler and an automatic HRR is issued.

Error: Busy Timeout

Busy timeout occurs if the VRB data buffer is not emptied out fast enough and the front end process cannot send data to the VRB.

This type of error can be triggered by several causes.

→ This type of error is detected by the error handler and an automatic HRR is issued.

Error: On Transition

A transition can fail if there is a problem initializing the front end electronics.

One typical error during a transition is: “Error Initializing HDI”.

In this case one has to try the coldstart transition again.

Calibrations

Calibrations for the different subsystems are also performed using Run Control.

The set of calibrations is part of the ace's duties.

Typically the software event builder is used for calibrations. This can accommodate larger event sizes.

Calibration data is sent to a *calibration consumer* which writes the results to a database.

Can view the results of the after being written into the database using **DBANA**.

Calibrations include:

Calorimeter - QIE, ShowerMax, LED and Xenon, Laser
CLC, COT, Silicon, Muon, TOF, BSC...

Details can be found at:

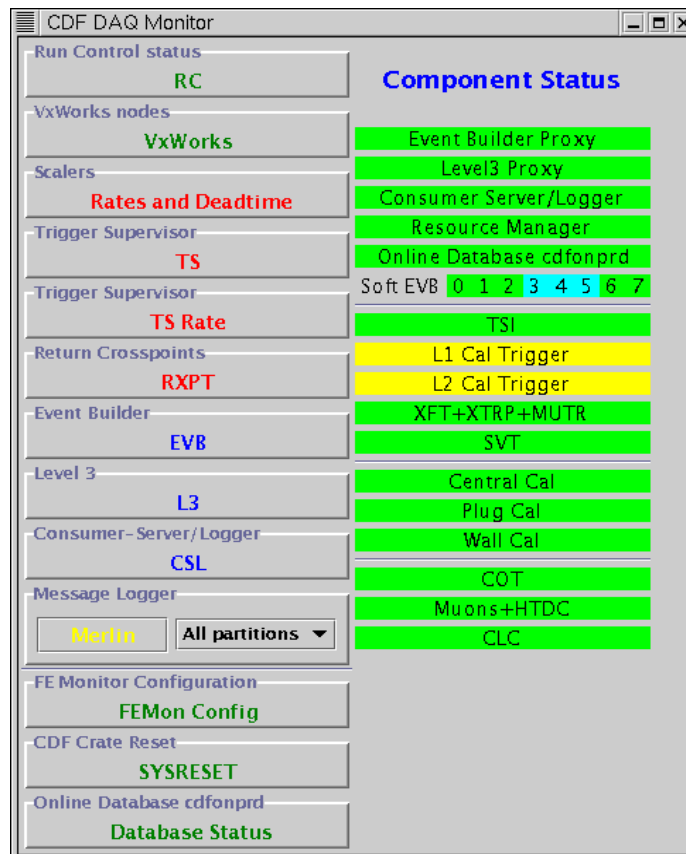
http://www-b0.fnal.gov:8000/ace2help/ace_calibrations.html

Also in a separate talk...

DAQ Monitoring

There are a number of programs that can be used to monitor the performance of the DAQ. Can be launched from a main control panel...

- > setup fer
- > daqmon



These monitors are mostly used to check that data is flowing through the system.

The quality of the data is checked by the consumers.

L3 Display

Shows all the L3 nodes (PCs running Linux)

Converters											
c01	c02	c03	c04	c05	c06	c07	c08				
1553	1536	1556	1539	1520	1547	1559	1591				
1553	1535	1550	1539	1525	1547	1558	1521				
31.5 Hz	31.5 Hz	30.4 Hz	31.5 Hz	30.4 Hz	31.5 Hz	30.4 Hz	30.4 Hz				
Processors											
001	010	015	051	067	033	000	115				
002	020	016	052	068	034	100	116				
003	021	017	053	060	035	101	117				
004	022	018	054	070	036	102	118				
005	023	010	055	071	037	103	119				
006	024	040	056	072	038	104	120				
007	025	041	057	073	039	105	121				
008	026	042	058	074	020	106	122				
009	027	043	059	075	021	107	123				
010	028	044	050	076	022	108	124				
011	029	045	051	077	023	109	125				
012	030	046	052	078	024	110	126				
013	031	047	053	079	025	111	127				
014	032	048	054	080	026	112	128				
015	033	049	055	081	027	113	129				
016	034	050	056	082	028	114	130				
Output nodes											
u01	6100	0	u02	6112	0	u03	6266	0	u04	6380	0
03.4 Hz	0.2 MB/s	03.8 Hz	0.2 MB/s	03.2 Hz	0.2 MB/s	03.2 Hz	0.2 MB/s				

⇒ More details in separate talk

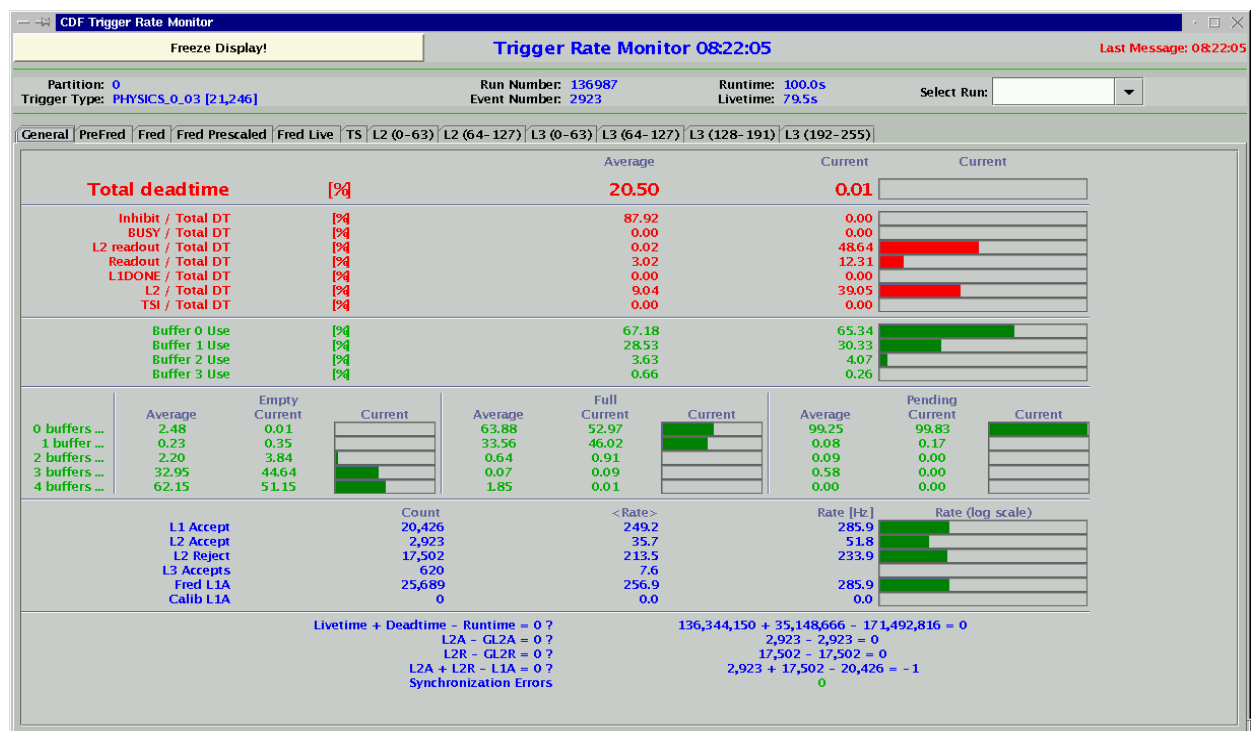
The more popular DAQ monitors are:

L3

Gives an overview of how L3 is working

Rates and Deadtime

General display...



Useful to check that the trigger is properly functioning.

Shows the total system **deadtime** and how it is allocated.

Several tabed panels are available to give you a detailed look at the rates for each trigger at L1, L2 and L3.

Use one of the tabs to look at specific trigger rates in more detail...

CDF Trigger Rate Monitor													
Freeze Display!				Trigger Rate Monitor 08:23:17						Last Message: 08:23:17			
Partition: 0				Run Number: 136987				Runtime: 175.8s				Select Run:	
Trigger Type: PHYSICS_0_03 [21,246]				Event Number: 7751				Livetime: 152.7s					
General	PreFired	Fred	Fred Prescaled	Fred Live	TS	L2 (0-63)	L2 (64-127)	L3 (0-63)	L3 (64-127)	L3 (128-191)	L3 (192-255)		
n	Trigger	Count	<Rate>	Rate, Hz	Rate (log scale)	n	Trigger	Count	<Rate>	Rate, Hz	Rate (log scale)		
0	L1_EM8 & CLC [1]	11,952	77.0	80.9	<div></div>	32	L1_CMUP6_PT4 [1]	316	2.0	9.3	<div></div>		
1	L1_CEM4_PT4 & CMX1.5_P...	69	0.4	2.1	<div></div>	33		0	0.0	0.0	<div></div>		
2	L1_EM8 & MET15 & CLC [3]	1,062	6.8	6.9	<div></div>	34		0	0.0	0.0	<div></div>		
3	L1_CEM4_PT4_PS25 [2]	7,624	49.1	111.4	<div></div>	35		0	0.0	0.0	<div></div>		
4		0	0.0	0.0	<div></div>	36		0	0.0	0.0	<div></div>		
5	L1_JET5_PS20 [2]	235,975	1,520.3	1,522.5	<div></div>	37		0	0.0	0.0	<div></div>		
6		0	0.0	0.0	<div></div>	38		0	0.0	0.0	<div></div>		
7	L1_TAU0_PT4_PS250 [2]	177,470	1,143.4	2,985.6	<div></div>	39		0	0.0	0.0	<div></div>		
8	L1_CEM8_PT8 [2]	484	3.1	6.3	<div></div>	40		0	0.0	0.0	<div></div>		
9	L1_TWO_CEM2_PT2_OPPQ...	5,476	35.3	110.8	<div></div>	41		0	0.0	0.0	<div></div>		
10		0	0.0	0.0	<div></div>	42		0	0.0	0.0	<div></div>		
11	L1_CMU0_PT4_PS250 [1]	170,800	1,100.4	2,856.5	<div></div>	43		0	0.0	0.0	<div></div>		
12		0	0.0	0.0	<div></div>	44		0	0.0	0.0	<div></div>		
13	L1_CMU6_PT8_PS1 [1]	62	0.4	2.6	<div></div>	45		0	0.0	0.0	<div></div>		
14		0	0.0	0.0	<div></div>	46		0	0.0	0.0	<div></div>		
15	L1_SEVEN_TRK2_PS50 [1]	514	3.3	14.3	<div></div>	47		0	0.0	0.0	<div></div>		
16	L1_JET10 & SUMET90 & CL...	486	3.1	3.7	<div></div>	48		0	0.0	0.0	<div></div>		
17	L1_TRK8_PS250 [1]	14,689	94.6	249.4	<div></div>	49		0	0.0	0.0	<div></div>		
18	L1_MET25 & CLC [1]	409	2.6	3.4	<div></div>	50		0	0.0	0.0	<div></div>		
19	L1_TWO_TRK2_OPPQ_DPHI1...	54,493	351.1	1,175.2	<div></div>	51		0	0.0	0.0	<div></div>		
20	L1_JET10 [1]	22,347	144.0	148.9	<div></div>	52		0	0.0	0.0	<div></div>		
21	L1_M8_CLC_PS200K [1]	28,001,548	180,405.5	180,449.6	<div></div>	53	L1_TRK4_PS1K [1]	167,614	1,079.9	2,814.2	<div></div>		
22	L1_TWO_CEM4_PT4 [2]	340	2.2	5.6	<div></div>	54		0	0.0	0.0	<div></div>		
23	L1_MB_XING_PS1M [2]	266,234,104	1,715,265.9	1,715,265.0	<div></div>	55	L1_TWO_TRK2_PS10K [1]	209,444	1,349.4	4,405.7	<div></div>		
24	L1_EM8 & CMU1.5_PT1.5 [1]	154	1.0	2.4	<div></div>	56		0	0.0	0.0	<div></div>		
25	L1_EM8 & CMX1.5_PT2 [1]	47	0.3	1.9	<div></div>	57		0	0.0	0.0	<div></div>		
26	L1_CMU1.5_PT1.5 & CMX1...	164	1.1	3.7	<div></div>	58		0	0.0	0.0	<div></div>		
27	L1_CMU6_PT4 [1]	1,626	10.5	30.7	<div></div>	59		0	0.0	0.0	<div></div>		
28	L1_TWO_CMU1.5_PT1.5 [2]	176	1.1	4.8	<div></div>	60		0	0.0	0.0	<div></div>		
29		0	0.0	0.0	<div></div>	61		0	0.0	0.0	<div></div>		
30		0	0.0	0.0	<div></div>	62		0	0.0	0.0	<div></div>		
31		0	0.0	0.0	<div></div>	63		0	0.0	0.0	<div></div>		

→ Normally the physics triggers are defined so that the dead time is less than about 5%.

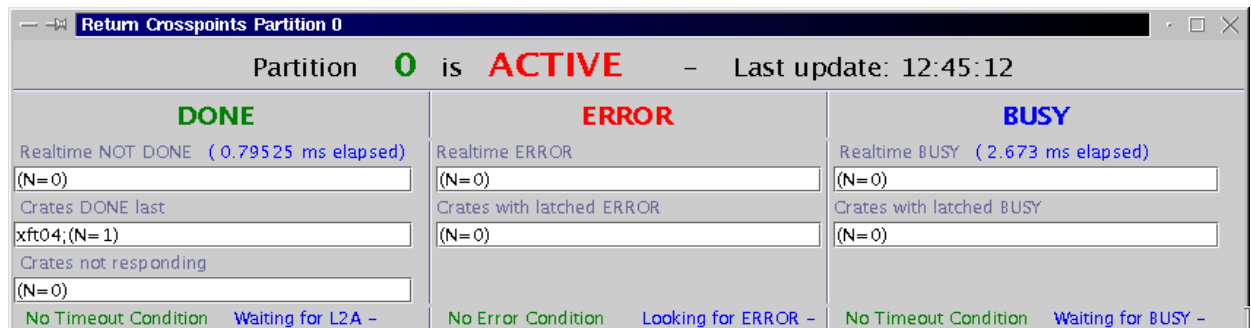
If you have a run going with high dead time this indicates a problem which needs to be fixed.

You can sometimes see which trigger is firing at a high rate from the "Rates and Deadtime" GUI.

RXTP

Shows which client was the last to return DONE, BUSY and ERROR and shows the time it took.

Useful to identify which crate is contributing to the deadtime.



In this example the XFT04 crate was the last to return a Done, and it was set 0.79 ms after the L2 decision was received.

→ Taking a long time to set the Done can result in "Readout Deadtime".

Typically a crate should set the done within 1 ms, but there are a few crates which can take longer.

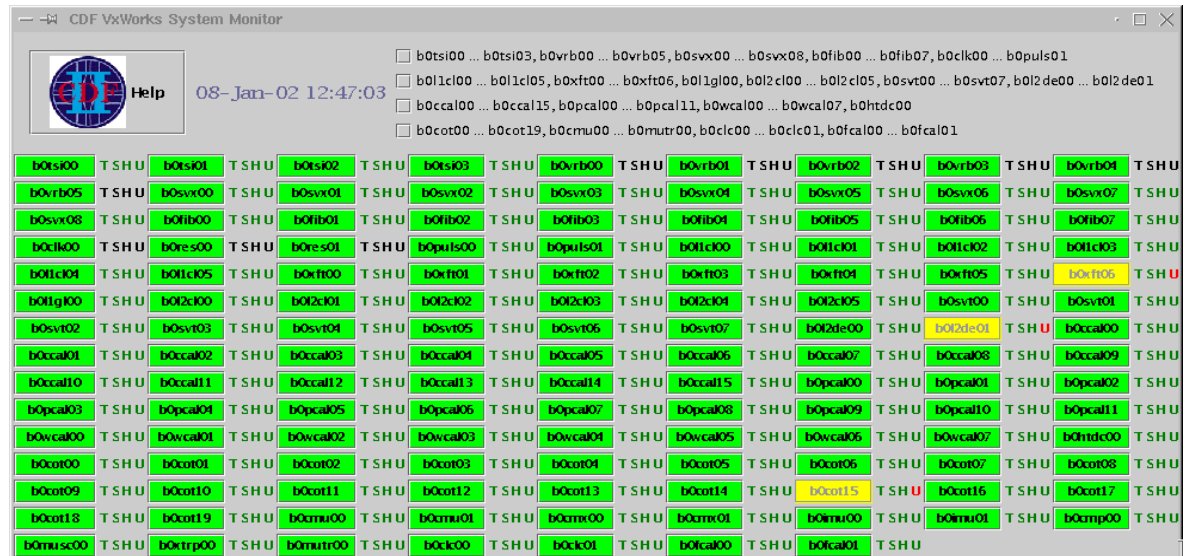
There is also a consumer (daqmon) that histograms the readout times and event sizes, more later...

CSL

Shows the status of the CSL, logging rates, partitions etc...

VxWorks Monitor

The VxWorks monitor gives an overview of the status of the Front End crates in the system.



Each button corresponds to one of the front end crates.

Green indicates that the process is OK

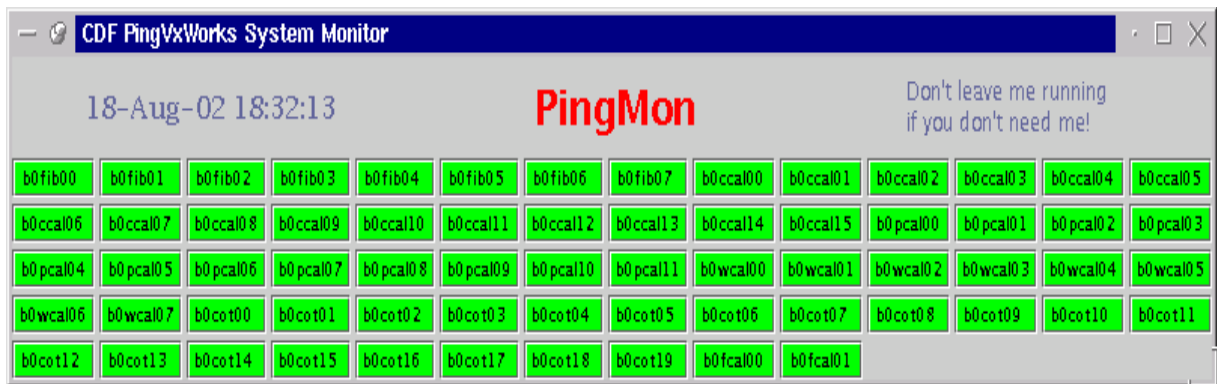
Yellow indicates that the crate is not updating information

Red indicates that problem with the crate

Useful to see if a front end crate has crashed...

PingMon

Ping to all CPUS in crates to check they reply



- setup fer
- pincmon

Consumers

Various consumers are used to check the quality of the data.

These are essentially AC++ modules compiled within the consumer framework used to monitor the quality of the data and the performance of the trigger.

Event Display

YMon

TrigMON

XMon

LumMon

Stage0

SiliMon

ObjectMon

BeamMon

L3RegionalMon

SVXMon

SVTMon

DAQMon

Used to identify hot channels (channels that are always on or are noisy), and dead regions (broken cables, high voltage problems...).

During shift operations there is a dedicated person (CO - Consumer Operator) assigned to look at the data quality.

Details in separate talk and at:

<http://www-b0.fnal.gov:8000/consumer/howto.html>

DAQMon Consumer

The DAQMon consumer plots the average readout time for each front end crate and the event size.

Noisy channels can sometimes lead to long readout times and large event fragments.



Important CDF DAQ Processes

Run control communicates with several key processes via so called proxy processes.

These processes are normally running but on rare occasions you may need to restart them.

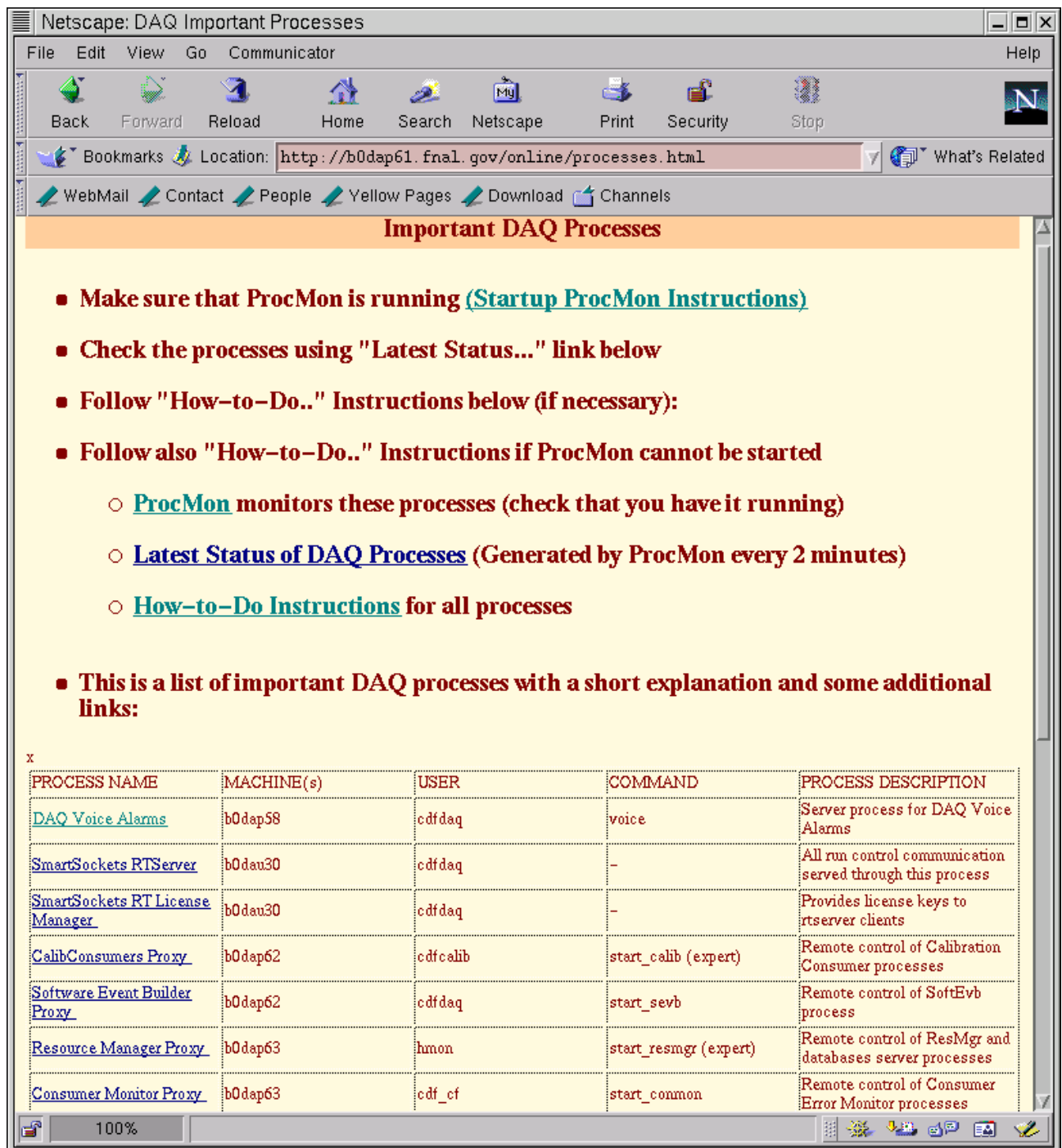
In addition to these there are a number of other essential processes, for a summary see the “Important CDF DAQ Processes” link from the ace help page where you can find instructions on starting the processes.

SmartSockets b0dau30
Calibration Consumer Proxy b0dap62
Software EVB Proxy b0dap62
Resource Manager b0dap63
Consumer Monitor Proxy b0dap63
DBbroker Proxy b0dap63
L3Manager b0dap31
ACNET Monitor b0dap68
SVTSPYMON b0dap68
Consumer Server Logger b0dau32
Calibration Consumer b0dap60
Partition 14 Sender b0dap60

In addition there are

Consumer Disk Server b0dap65
Silicon Disk Server b0dap41

Important DAQ Processes



Important DAQ Processes

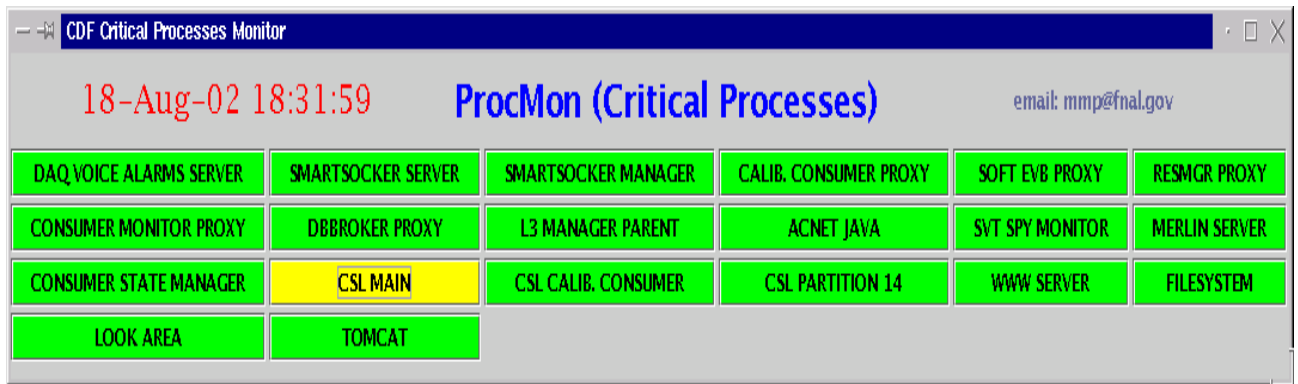
- **Make sure that ProcMon is running ([Startup ProcMon Instructions](#))**
- **Check the processes using "Latest Status..." link below**
- **Follow "How-to-Do.." Instructions below (if necessary):**
- **Follow also "How-to-Do.." Instructions if ProcMon cannot be started**
 - [ProcMon](#) monitors these processes (check that you have it running)
 - [Latest Status of DAQ Processes](#) (Generated by ProcMon every 2 minutes)
 - [How-to-Do Instructions](#) for all processes
- **This is a list of important DAQ processes with a short explanation and some additional links:**

PROCESS NAME	MACHINE(s)	USER	COMMAND	PROCESS DESCRIPTION
DAQ Voice Alarms	b0dap58	cdftdaq	voice	Server process for DAQ Voice Alarms
SmartSockets RTServer	b0dau30	cdftdaq	-	All run control communication served through this process
SmartSockets RT License Manager	b0dau30	cdftdaq	-	Provides license keys to rtserver clients
CalibConsumers Proxy	b0dap62	cdftcalib	start_calib (expert)	Remote control of Calibration Consumer processes
Software Event Builder Proxy	b0dap62	cdftdaq	start_sevb	Remote control of SoftEvb process
Resource Manager Proxy	b0dap63	hmon	start_resmgr (expert)	Remote control of ResMgr and databases server processes
Consumer Monitor Proxy	b0dap63	cdftcf	start_common	Remote control of Consumer Error Monitor processes

List of critical DAQ Processes + instructions
ProcMon startup instructions

ProcMon Monitor (CDF-note 6045)

ProcMon performs online monitoring of several crucial software processes in the CDF DAQ online system.



It loops every ~ 2 minutes and checks sequentially (one-by-one) the different processes.

- Yellow: being tested
- Green : OK
- Red : process missing (or machine unreachable)
- Pink : multiple copies of same process

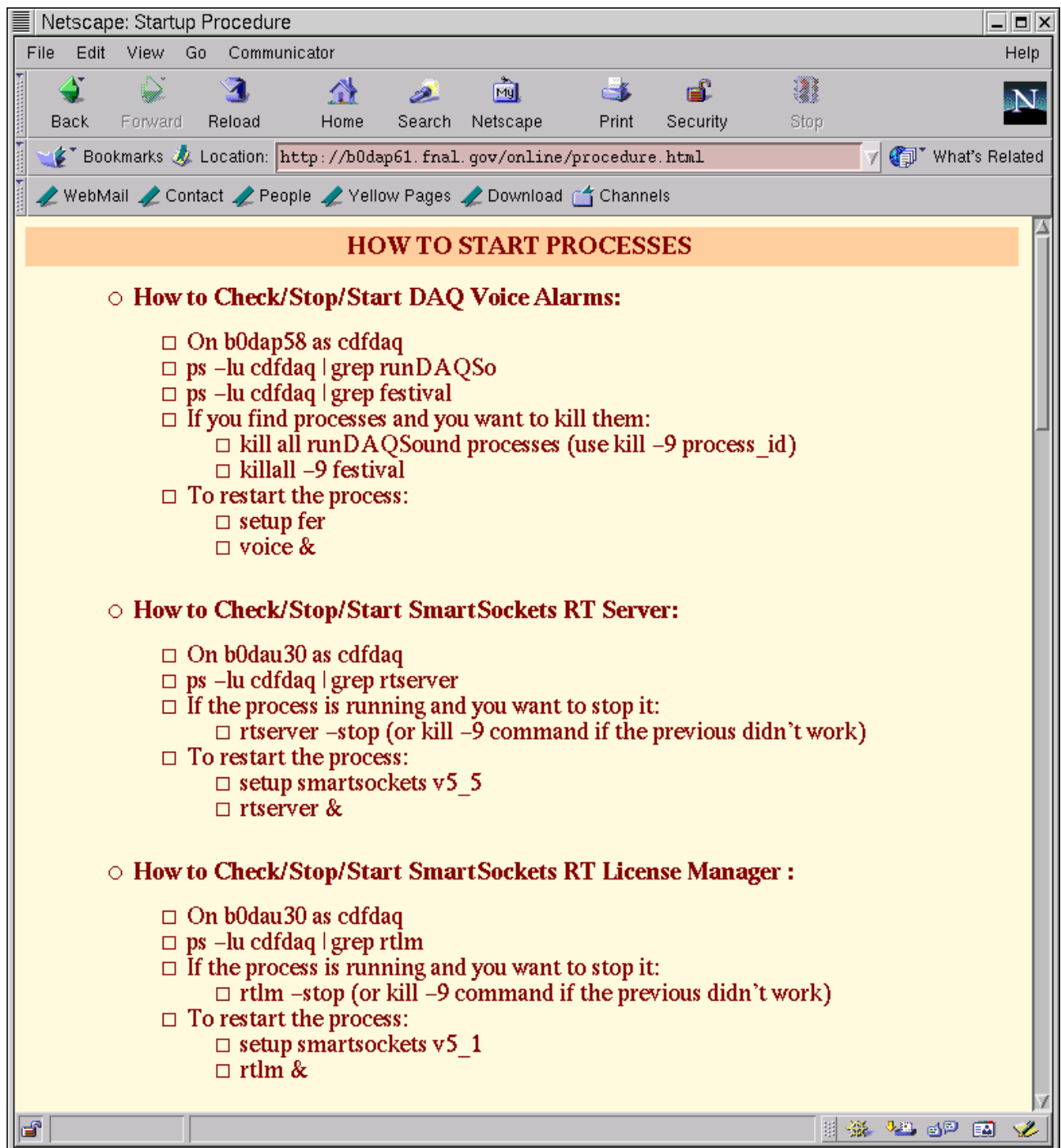
In case of problems:

- you get error box + voice alarm
- Follow instructions!

- ⇒ These are Critical Processes for DAQ
- ProcMon must be running!
- Check ProcMon time (on display)
- Do not ignore ProcMon errors!

What to do When....

In case of ProcMon errors check the www page:

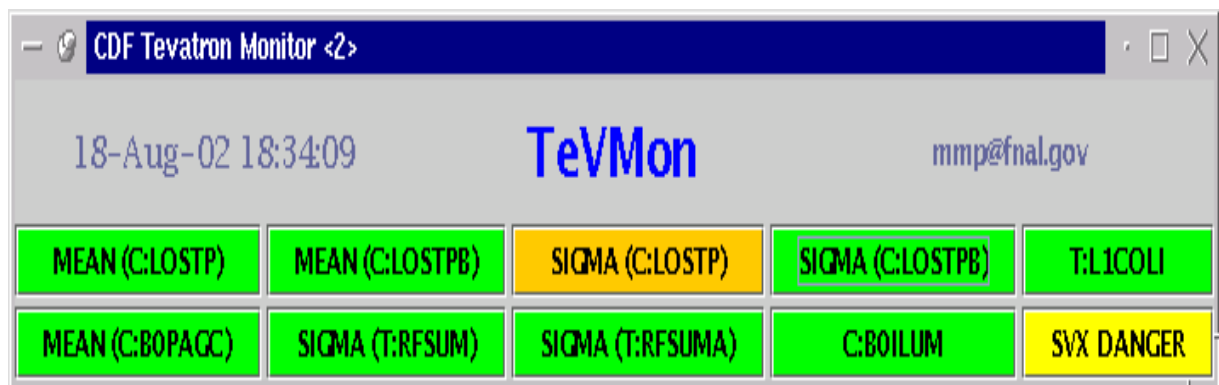


⇒ There you find precise instructions...

TeVMon (Monitors Beam Conditions)

Still under constructions but already running at B0

Monitors the beam conditions (losses, spikes, TEL, ...)



⇒ Produces an error in case of danger for silicon

⇒ Plan to use it in the future to decide whether to ramp up the HVs (coming soon...)

⇒ You will learn more about it during your shifts

Final Remarks

(the goal is to reach maximum efficiency)

- To be an ACE is an important job in CDF
 - You will get an unique overview of CDF operation
 - ...but also it will require full attention
 - Be focused (your laptop will not help you on that)
 - Sometimes SciCo's are real expertssometimes no
-
- Read e-logs from previous shifts
 - Check your e-mail for possible updates
 - Keep full control of resources (who has which crate)
 - Ask experts to document what they did on the e-logs
 - Try to document all problems in e-log precisely
-
- In case of problems consult information on the www
 - If you realize you cannot fix it...page an expert!
-
- Consider yourself a simple user of the online cluster
 - Don't play around with online PCs settings
 - Don't reboot online machines (contact experts)
 - Don't download any software
 - Don't allow non-authorized use